Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 48 (2015) 198 – 203

Procedia Computer Science

International Conference on Intelligent Computing, Communication & Convergence

(ICCC-2015)

Conference Organized by Interscience Institute of Management and Technology,

Bhubaneswar, Odisha, India

Enhancing security in Cloud using Trusted Monitoring Framework

M.Arun Fera[a], C.manikandaprabhu[b], Ilakiya Natarajan[c], K.Brinda[d], R.Darathiprincy[e] *

[a]*Assistant Professor Dept of IT, Thiagarajar College of engineering, Madurai, 625015, India*
[b,c,d,e]*Final year Dept of IT, Thiagarajar College of engineering, Madurai, 625015, India*

Abstract

Cloud computing is a technology that provides network based services on demand. Cloud computing technology provides advantages to end users and business organizations. Few notable advantages are cost efficiency, increased storage capacity, backup and recovery, continuous resource availability and location independence. Data owners host their private data in the cloud and worry about unauthorized access of their data. They feel uncomfortable about any user misusing their private data. This insecure feeling of data owners holds them back from using cloud services. Any unauthorized users accessing the owner's private data leads to accountability issues. We design a trusted monitoring framework, which provides a chain of trust that excludes the untrusted privileged domain, as well as utilizing the trusted computing technology to ensure the integrity of the monitoring environment. To solve the accountability issue, a mechanism to monitor the actual data usage is proposed. This approach grants access rights to users based on their role and also monitors every access to the owner's data, verifying that the service level agreements have been violated or not.

Keywords: Cloud computing, trusted monitoring framework, Security, Chain of trust, Data auditability.

Peer-review under responsibility of scientific committee of International Conference on Computer, Communication and Convergence (ICCC 2015)

* Corresponding author. Tel.: +919600302361;
*E-mail address:*cmanikandaprabhuc@gmail.com

Peer-review under responsibility of scientific committee of International Conference on Computer, Communication and Convergence (ICCC 2015)
doi:10.1016/j.procs.2015.04.170

1. Introduction

Cloud Computing technology provides advantages to end users and business organizations. Few notable advantages are cost efficiency, increased storage capacity, backup and recovery, continuous resource availability and location independence. In spite of these advantages, the biggest issue with the cloud is the "Security". Though there are several advantages with cloud, it also imposes several security threats related to outsourced user's data. Since private data is hosted in the cloud and they are being processed at remote machines and are administered by the cloud service providers (CSP), the users are worried about loss of data control in the cloud. There are various reasons for the CSP to involve in unfaithful disclosure or leakage of user's data to any external entity that may turn out to be a serious privacy and security concern for any user towards his/her data. Cloud Users do not know the machines where their data are processed, so they start bothering about losing control over their data. There are no specific mechanisms to check if the service level agreements made between the data owner and the end users have been preserved or not. Data is often being outsourced in cloud, leading to accountability issues and manipulation of personally identifiable information. The monitoring mechanism involves third party services. The third party is an external entity who can behave unfaithfully while the data is disclosed during the auditing process [1]. Moreover, simply relying on a third party auditor without applying any cryptographic technique on the user's data may turn the situation even more badly. So, downloading the user's data alone for auditing will not help for verifying the integrity of user's data since downloading the entire data is expensive because of I/O and transmission cost through the network. To solve these security problems with the cloud, users are assigned access rights based on their role and if any user tries to violate the assigned access rights the data owner is notified with a log report stating about the service level agreement violation. Section 2 tells the related work done in this domain, section 3 explains the work that is done for solving the security issues, section 4 describes the results and the discussion among the results and section 5 concludes the paper.

2. Related work

Cloud Computing is a technology that provides network based services on-demand. Data owners host their private data in the cloud and worry about unauthorized access of their data [2]. They feel uncomfortable about any user misusing their private data. This insecure feeling of data owners holds them back from using cloud services. Any unauthorized users accessing the owner's private data leads to accountability issues. To solve the accountability issue, a mechanism to monitor the actual data usage is proposed. This approach grants access rights to users based on their role and also monitors every access to the owner's data, verifying that the service level agreements have been violated or not.

Though there are several advantages with cloud, it also imposes several security threats related to outsourced user's data. Since private data is hosted in the cloud and they are being processed at remote machines and are administered by the cloud service providers (CSP), the users are worried about loss of data control in the cloud. There are various reasons for the CSP to involve in unfaithful disclosure or leakage of user's data to any external entity that may turn out to be a serious privacy and security concern for any user towards his/her data [3].

Virtualization is a pillar technology in cloud computing for multiplexing computing resources on a single cloud platform for multiple cloud tenants. Monitoring the behaviour of virtual machines (VMs) on a cloud platform is a critical requirement for cloud tenants. Existing monitoring mechanisms on virtualized platforms either takes a complete VM as the monitoring granularity, such that they cannot capture the malicious behaviours within individual VMs, or they focus on specific monitoring functions that cannot be used for heterogeneous VMs concurrently running on a single cloud node. Furthermore, the existing monitoring mechanisms have made an assumption that the privileged domain is trusted to act as expected, which causes the cloud tenants' concern about security because the privileged domain in fact could not act as the tenants' expectation. We design a trusted monitoring framework, which provides a chain of trust that excludes the untrusted privileged domain, by deploying an independent guest domain for the monitoring purpose, as well as utilizing the trusted computing technology to ensure the integrity of the monitoring environment.

The user is more concerned with the privacy and security issues in the cloud environment. Ina cloud platform, the

data that belongs to the end user is stored in the servers in the cloud service provider premises [4]. The most important concern of the end user is that whether the sensitive information would be leaked so that hackers would get to know about the confidential information. The authors focussed on the development of a more secure cloud environment, to find the the trust of the service requesting authorities by using a new VM (Virtual Machine) monitoring system [4]. The authors claim that the framework can be used to provide security in network, infrastructure and in a heterogeneous cloud platform. It provides storage security by monitoring unauthorized access activities by the Cloud Service Users (CSU). Infrastructure security is provided by monitoring the use of privileged instructions.

Cloud computing is a model for enabling service user's ubiquitous, convenient and on-demand network access to a shared pool of configurable computing resources. The security for Cloud Computing is emerging area for study and this paper provide security topic in terms of cloud computing based on analysis of Cloud Security treats and Technical Components of Cloud Computing [5].

Cloud computing is a model for enabling service user's ubiquitous, convenient and on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services), that can be rapidly provisioned and released with minimal management effort or service provider interaction. Cloud computing enables cloud services. The security architecture and functions highly depend on the reference architecture [6]. A key differentiating element of a successful information technology (IT) is its ability to become a true, valuable, and economical contributor to cyber infrastructure. "Cloud" computing embraces cyber infrastructure, and builds upon decades of research in virtualization, distributed computing, "grid computing", utility computing, and, more recently, networking, web and software services.

3 Proposed work

3.1 Cloud Data Accountability Framework

The major components of the CDA framework are the log generator [7] [8]and the log converger. The log generator is responsible for the generation of log records for every JAR access and the log converger is responsible for decrypting logs for error correction, merging and sending logs to the data owner. The logs are encrypted using the public key generated by the data owner and the logs are decrypted by the data converger using the master key for integrity verification of logs. The data owner creates a JAR file that encloses the original encrypted data, the access control policies and the logging policies. These access control policies help the CSP in authenticating the user and also in granting access rights to the users. These access control policies are fixed by the data owners based on the user's role to be played in accessing the cloud data. The users are authenticated by the cloud service provider and the requested JAR file access will be granted to the user based on the previously specified access control rights based on user's role. The automated logging mechanism is initiated for every data access and the log generator begins to generate log records for every data access in the cloud [9][10]. The log generator is a nested JAR file storing user data and respective log files. The outer JAR contains access control policies for each user based on their role and is responsible for user authentication for a particular JAR file based on the specified access rights. The outer JAR may contain one or more inner JARs. The outer JAR helps in identifying the correct inner JAR based on user request. The inner JAR consists of the data in encrypted form, class files initiating log generation, displaying data in required format and generation of log file for each encrypted data. The two auditing modes help the data owner to monitor the usage of his/her data hosted in the cloud. In Automatic mode, the size of the log record and the maximum time that should elapse before dumping of log files are fixed by the data owner. If any one of these condition occurs, logs are being sent to the data owner. In Onrequest mode, the onrequest mode can be used by the data owners, if he/she suspects of data being misused. This mode helps the owner to monitor the data usage immediately. Possible attacks to the CDA framework are discussed here. This framework encounters the following security attacks and the solution for the same is provided here. The Jar copy attack copies the JAR files and assumes that it will allow him to access the JAR file data without being noticed by the data owner. But every access to a JAR file creates log record and being sent to the data owner for auditing. Even for additional copies of the JAR, log records will be generated and sent to the data owner. If any copy of a JAR file is moved to a location that could not be accessed by the log converger, access to that particular JAR file is cancelled. The JAR file becomes inaccessible. Disassembling attack is another attack possible in our scenario. Once the Jar files have been obtained by the hacker, he/she tries to disassemble the JAR file. But the JAR file contains data and logs in encrypted format. The hacker

knows only the public key used for encrypting the logs, there are no possibilities to obtain the master key used for decrypting the logs. The attacker cannot modify the log file content after disassembling the JAR. The integrity checking mechanism for logs will detect any modification in the log records since the integrity checks added to each record will not match at the time of verification by the log converger. The Reed-Solomon encoding is for this checking mechanism using which the log converger can easily detect corrupted logs.

3.7 Design

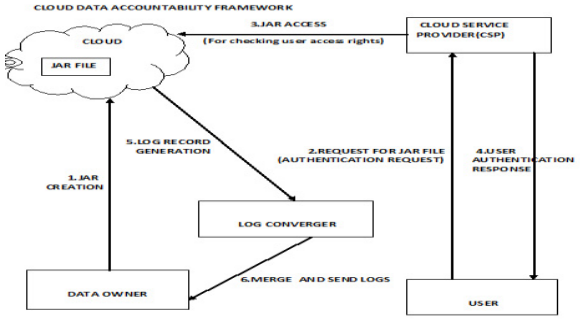

Fig 1 Cloud data accountability framework

Figure 1 tells the design of the entire system where a jar file is created with all kinds of rights of the users who are registered. It also explains the way by which the cloud user could interact with the system, where the user rights are granted only if the rule is present in the protected jar file. Figure 2 tells the structure of the jar file.

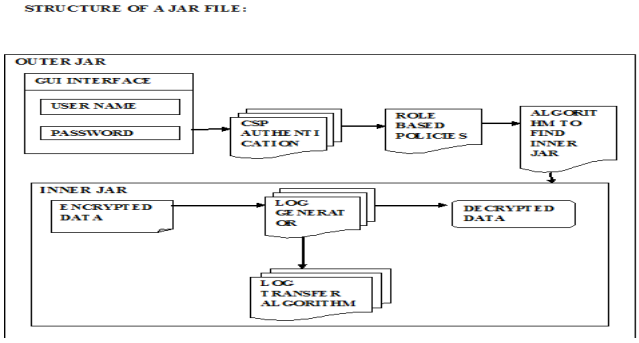

Fig 2 Structure of the JAR File

4 Results and Discussion

4.1 Solution to Security issue

In order to solve the security issue [11], we separate the semantic reconstruction functionality out from VMM level. We propose the monitoring driver, which encapsulates the semantic reconstruction that will process the OS-related information of a monitored user VM. The monitoring driver is dynamically loaded when the corresponding VM is launched on or migrated to this platform. Once the issue is resolved, the users could use this with trust [12]. Whenever an event is intercepted by event sensor, the low level information is intercepted and its semantic information is reconstructed by the driver. Thus, the monitoring applications in the monitoring VM can be independent of guest OS. Through this way, the monitoring framework separates the monitoring point from the semantic reconstruction module and masks the variance of guest OS for general but fine-grained monitoring

mechanism. In order to solve the trust issue, we separate the monitoring functionality out of privileged domain, and put it into an independent domain which is specially designed for the monitoring task for the tenants. By this way, we can achieve the separation of privilege.

4.2 Chain of trust

To avoid the security issue caused by the distrusted privileged domain (management VM, i.e., MVM), there should be a new construction approach for the ‘‘chain of trust’’, so that the MVM is not able to modify the monitoring data. Monitoring VM acquires the monitoring data through semantic reconstruction, and the monitoring data should be guaranteed to be integrity. Only with this, the trust issue between the cloud tenants and the privileged domains can be solved. We see the adoption of trusted computing technology is a promising way for establishing trust, but how to construct the ‘‘chain of trust’’, and how to accommodate multiple tenants into multiple trust chains is up to be solved.

The goal of the adoption of trusted computing into the monitoring framework is not to make trusted computing functionalities be available to the cloud infrastructure (i.e., the user VMs) [13], but to make the monitoring framework itself is able to utilize the trusted computing functionalities. Therefore, there are some differences between the scenario in our monitoring framework and the scenario in the original case. We use vTPM to ensure that the tenants can establish trust in the environment where the monitoring information is hosted (i.e., the monitoring VM, and the OS and specific application for monitoring running in it), while vTPM is used to ensure that the tenants can establish trust in the environment which consists of the tenanted VMs, and the OSes and applications running in it [14][15].

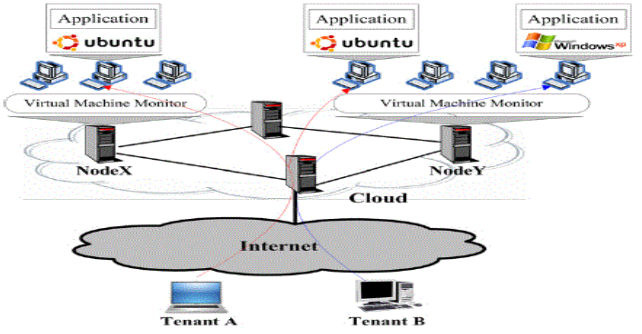

Fig 3 An example of cloud computing platform

Figure 3 tells the implementation framework which has virtual machines running the images of ubuntu and windows. The virtual machine monitor or the hypervisor manages the creation and deletion of virtual machines.

Table 1 Characteristics of virtualization based monitoring solutions

	Monitoring granularity	Monitoring generality	Transparency to guest OS	Integrity of monitoring entity
Performance monitoring	Whole VM	Good	Yes	No
Passive VMI	Fine grained	OS type dependency	Yes	No
Active VMI	Fine grained	OS version dependency	No	No
Our solution	Fine grained	Good	Yes	Yes

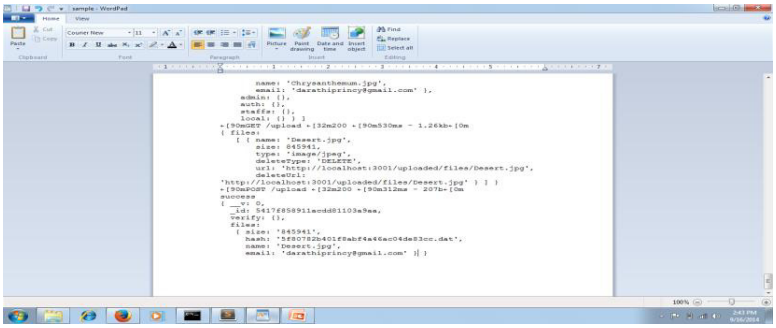

Fig 4 Log Report

5 Conclusion and Future work

Thus the Trusted Monitoring Framework has been successfully implemented in the cloud environment. It enhances the security against the intruders and the cloud provider. It generates a log for the data that are being accessed from the cloud and provides the encrypted log to the data owner. The future aim of this project is to create various VMs for various cloud users thereby enhancing the security of the cloud and by using various security algorithm to protect the data.

References

1. Smitha Sundareswaran, Anna C. Squicciarini and Dan Lin, "Ensuring Distributed Accountability for Data Sharing in the Cloud,", IEEE Transaction on dependable a secure computing, VOL. 9, NO. 4, pg 556-568, 2012.
2. S. Sundareswaran, A. Squicciarini, D. Lin, and S. Huang, "Promoting Distributed Accountability in the Cloud," *Proc. IEEE* Int‟l Conf. Cloud Computing, 2011.
3. Ryan K L Ko, Peter Jagadpramana, Miranda Mowbray, Siani Pearson, Markus Kirchberg, Qianhui, "TrustCloud: A Framework for Accountability and Trust in Cloud Computing" HP Laboratories, pp 1 – 7, HPL-2011.
4. A. Squicciarini , S. Sundareswaran and D. Lin, " Preventing Information Leakage from Indexing in the Cloud," *Proc. IEEE Int'l* Conf. Cloud Computing, 2010.
5. S. Pearson , Y. Shen, and M. Mowbray," A privacy Manager for Cloud Computing," Proc. Int'l Conf. Cloud Computing (cloudcom), pp.90-106,2009.
6. S. Pearson and A. Charlesworth, "Accountability as a Way Forward for Privacy Protection in the Cloud, " *Proc First Int'l conf. Cloud* Computing, 2009.
7. R. Corin, S. Etalle, J.I. den Hartog, G. Lenzini, and I. Staicu, "A Logic for Auditing Accountability in Decentralized Systems," Proc. IFIP TC1 WG1.7 Workshop Formal Aspects in Security and Trust, pp. 187-201, 2005.
8. B. Chun and A. C. Bavier ,"Decentralized Trust Management and Accountability in Federated System," *Proc. Ann. Hawaii Int'l Conf. System Science (HICSS), 2004.*
9. B. Crispo and G. Ruffo, "Reasoning about Accountability within Delegation," Proc. Third Int‟l Conf. Information and Comm. Security (ICICS), pp. 251-260, 2001.
10. D.J. Weitzner, H. Abelson, T. Berners-Lee, J. Feigen-baum, J. Hendler, and G.J. Sussman, "Information Accountability," Comm. ACM, vol. 51, no. 6, pp. 82-87, 2008.
11. M. Armbrust, A. Fox, R. Griffith, A. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, M. Zaharia, Above the clouds: a berkeley view of cloud computing, Technical Report, EECS Department, University of California, Berkeley, 2009.
12. R. Buyya, C.S. Yeo, S. Venugopal, J. Broberg, I. Brandic, Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility, Future Generation Computer Systems 25 (6) (2009) 599–616.
13. S. Nanda, T. Chiueh, A survey on virtualization technologies, Technical Report, ECSL-TR-129, SUNY at Stony Brook, 2005.
14. J. Li, B. Li, T. Wo, C. Hu, J. Huai, L. Liu, K.P. Lam, CyberGuarder: a virtualization security assurance architecture for green cloud computing, Future Generation Computer Systems 28 (2) (2012) 379–390.
15. B.D. Payne, M. Carbone, M. Sharif, W. Lee, Lares: an architecture for secure active monitoring using virtualization, in: Proc. of the 2008 IEEE Symposium on Security and Privacy, 2008, pp. 233–247.